



①9 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENTAMT

⑫ **Offenlegungsschrift**
⑩ **DE 197 13 178 A 1**

⑤1 Int. Cl.⁶:
G 06 F 12/08
G 06 F 13/16

②1 Aktenzeichen: 197 13 178.6
②2 Anmeldetag: 27. 3. 97
④3 Offenlegungstag: 1. 10. 98

⑦1 Anmelder:
Siemens AG, 80333 München, DE

⑦2 Erfinder:
Knuth, Robert, Dr., 81737 München, DE

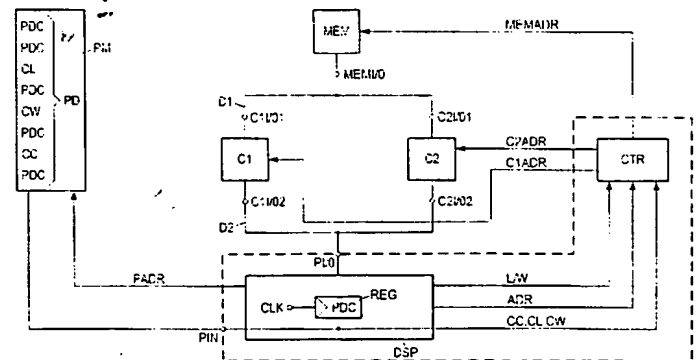
⑤6 Entgegenhaltungen:
DE 43 30 751 A1
EP 04 96 439 A2

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

⑤4 Schaltungsanordnung mit einem Prozessor und einem Datenspeicher

⑤7 Die Schaltungsanordnung weist zwischen einem Prozessor (DSP) und einem zugehörigen Datenspeicher (MEM) zwei parallel geschaltete Cache-Speicher (C1, C2) auf, aus denen jeweils wechselseitig ausgelesen werden kann, während der jeweils andere Cache-Speicher zeitgleich mit Daten aus dem Datenspeicher (MEM) nachladbar ist. Hierdurch ist der Datenfluß verstetigbar und für ihn herrschen Echtzeitbedingungen. Besonders geeignet für den Einsatz von digitalen Signalprozessoren.



DE 197 13 178 A 1

DE 197 13 178 A 1

Die Erfindung betrifft eine Schaltungsanordnung mit einem Prozessor und einem Datenspeicher. Im Datenspeicher sind durch den Prozessor zu verarbeitende Daten gespeichert. Dies können entweder Programmbefehle oder Operanden sein.

Es ist bekannt, zur Anpassung der Datenrate eines relativ langsamen Datenspeichers an die im Vergleich dazu höhere Verarbeitungsgeschwindigkeit des Prozessors sogenannte Cache-Speicher einzusetzen. Diese dienen zur Zwischenspeicherung von im Datenspeicher gespeicherten Daten, bei denen die Wahrscheinlichkeit relativ hoch ist, daß der Prozessor in der nächsten Zeit auf sie zugreift. Als Datenspeicher kommen beispielsweise häufig DRAMs (dynamische Speicher mit wahlfreiem Zugriff) zum Einsatz, die relativ kostengünstig sind, jedoch auch relativ langsam. Als Cache-Speicher werden dagegen beispielsweise SRAMs (statische Speicher mit wahlfreiem Zugriff) eingesetzt, die gegenüber DRAMs zwar großflächiger und teurer, aber andererseits auch wesentlich schneller sind.

Zum Betrieb der geschilderten Schaltungsanordnung ist eine Cache-Steuerung notwendig, die Datenanforderungen des Prozessors erhält und daraufhin den Cache-Speicher so adressiert, daß die nachgefragten Daten an den Prozessor ausgegeben werden. Die Datenanforderungen resultieren aus aktuell vom Prozessor zu verarbeitenden Programmbefehlen. Stellt die Cache-Steuerung fest, daß die aktuell nachgefragten Daten sich nicht im Cache-Speicher befinden, handelt es sich um einen sogenannten Cache-Miss. In diesem Fall muß die Cache-Steuerung die angeforderten Daten erst vom Datenspeicher in den als Zwischenspeicher fungierenden Cache-Speicher nachladen und gleichzeitig die Arbeit des Prozessors, beispielsweise durch Deaktivierung seines Arbeitstaktes, unterbrechen (sogenanntes "Anhalten" des Prozessors). Hierzu ist die Cache-Steuerung in der Lage, nicht nur den Cache-Speicher, sondern auch den Datenspeicher zu adressieren. Gleichzeitig wird nicht nur das aktuell vom Prozessor nachgefragte Datum vom Datenspeicher in den Cache-Speicher übertragen, sondern weitere Daten, bei denen die Wahrscheinlichkeit relativ hoch ist, daß der Prozessor demnächst auf sie zugreifen wird. Dies können beispielsweise Daten sein, die an auf die Adresse des aktuell angeforderten Datums nachfolgenden Adressen im Datenspeicher abgelegt sind. Letzteres trifft insbesondere zu, wenn die im Datenspeicher gespeicherten Daten Programmbefehle sind, da mit der Ausnahme von Sprungbefehlen Programme üblicherweise seriell in Richtung aufsteigender Adressen vom Prozessor abgearbeitet werden.

Ist nach einem aufgetretenen Cache-Miss der Cache-Speicher wieder mit neuen Daten geladen worden, kann die Cache-Steuerung erneut die Datenübertragung zwischen Cache-Speicher und Prozessor starten, so daß der Prozessor seine zuvor unterbrochene Arbeit mit der weiteren Verarbeitung des zuletzt geladenen Programmbefehls wieder aufnehmen kann. Bei einem erneuten Cache-Miss wiederholt sich der beschriebene Ablauf.

Außer dem oben beschriebenen Lese-Caching ist auch ein Schreib-Caching möglich. Dabei übermittelt der Prozessor diejenigen Adressen an die Cache-Steuerung, an denen von ihm erzeugte Daten im Datenspeicher MEM gespeichert werden sollen. Die Cache-Steuerung bewirkt dann eine entsprechende Übertragung dieser Daten vom Prozessor über den Cache-Speicher in den Datenspeicher. Bevor dieser gesamte Vorgang nicht abgeschlossen ist, kann kein Lesen von Daten durch den Prozessor erfolgen.

Insbesondere bei digitalen Signalprozessoren (DSP) ist es wünschenswert, daß der Datenfluß zwischen dem Daten-

speicher und dem Prozessor kontinuierlich erfolgt. Beispielsweise bei Einsatz von Embedded DRAMs als Datenspeicher ist die Zugriffsgeschwindigkeit beim Lesen aber in Abhängigkeit von den Adressen unterschiedlich groß (sogenannter Burst-Zugriff). Zusätzlich besteht für die Speicheranbindung in DSP-Systemen die wichtige Anforderung der Echtzeitfähigkeit. Das bedeutet, daß die Programmabarbeitung nicht unvorhersehbar lange unterbrochen werden darf.

Der Erfindung liegt die Aufgabe zugrunde, eine Schaltungsanordnung mit einem Prozessor und einem zugehörigen Datenspeicher anzugeben, bei der der Datenfluß zwischen Speicher und Prozessor weitestgehend kontinuierlich erfolgt.

Diese Aufgabe wird durch eine Schaltungsanordnung gemäß Anspruch 1 gelöst. Weiterbildungen und Ausgestaltungen der Erfindung sind in Unteransprüchen gekennzeichnet.

Erfindungsgemäß weist die Schaltungsanordnung zwei Cache-Speicher auf, die als Zwischenspeicher für die vom Datenspeicher zum Prozessor (und ggf. umgekehrt) zu übertragenden Daten dienen. Hierzu sind erste Anschlüsse der beiden Cache-Speicher über einen ersten Datenbus mit Anschlüssen des Datenspeichers und zweite Anschlüsse der Cache-Speicher über einen zweiten Datenbus mit den Anschlüssen des Prozessors verbunden. Die beiden Cache-Speicher sind also bezüglich aller Anschlüsse sowohl des Datenspeichers als auch des Prozessors parallel geschaltet, so daß vom Prozessor vorteilhafterweise abwechselnd aus ihnen ausgelesen oder in sie hineingeschrieben werden kann. Ebenso können auch abwechselnd Daten zwischen dem Datenspeicher und den Cache-Speichern übertragen werden.

Eine Ausführungsform der Erfindung sieht vor, daß bei wenigstens einem der Cache-Speicher die ersten und zweiten Anschlüsse wenigstens teilweise identisch sind. Es handelt sich bei einem solchen Cache-Speicher um einen Single-Port-Speicher.

Nach einer Weiterbildung der Erfindung ist es vorgesehen, daß jeweils abwechselnd Daten über den zweiten Datenbus zwischen einem der Cache-Speicher und dem Prozessor übertragen werden, während gleichzeitig Daten über den ersten Datenbus zwischen dem Datenspeicher und dem anderen Cache-Speicher übertragen werden. Dies hat den Vorteil, daß beispielsweise einer der Cache-Speicher bereits mit im Datenspeicher gespeicherten Daten beschreibbar ist, die der Prozessor erst benötigt, nachdem er Daten aus dem anderen, derzeit aktuell auszulesenden Cache-Speicher verarbeitet hat. Insgesamt bietet die Erfindung die Möglichkeit, über den ersten Datenbus in den einen Cache-Speicher zu schreiben oder aus ihm zu lesen und gleichzeitig über den zweiten Datenbus in den anderen Cache-Speicher zu schreiben oder aus ihm zu lesen. Daher ist es z. B. möglich, entweder den einen Cache-Speicher aus dem Datenspeicher nachzuladen und gleichzeitig aus dem anderen Cache-Speicher Daten zum Prozessor zu übertragen oder auch aus dem einen Cache-Speicher Daten in den Datenspeicher zu übertragen und gleichzeitig durch den Prozessor Daten aus dem anderen Cache-Speicher zu lesen oder in den anderen Cache-Speicher Daten einzuschreiben.

Eine Weiterbildung der Erfindung sieht vor, daß die Schaltungsanordnung eine Cache-Steuerung und einen Programmspeicher aufweist, wobei die Cache-Steuerung einer Steuerung der Datenübertragung auf den beiden Datenbussen dient und im Programmspeicher Programmdaten mit durch den Prozessor abzuarbeitenden Programmbefehlen gespeichert sind. Dabei ist vorteilhafterweise vorgesehen, daß die Programmdaten außer den Programmbefehlen wenigstens einen Cache-Ladebefehl beinhalten, der an die Cache-Steuerung übermittelbar ist und der das Laden eines der

Cache-Speicher mit zu einem späteren Zeitpunkt aufgrund eines der Programmbefehle vom Prozessor benötigten Daten aus dem Datenspeicher über den ersten Datenbus bewirkt. Dies hat den Vorteil, daß bereits Daten durch den Cache-Ladebefehl sozusagen "vorbestellt" werden, die aktuell noch gar nicht vom Prozessor benötigt werden. Sobald der Prozessor sie aber später benötigt, liegen sie bereits im Cache-Speicher vor. Durch die Cache-Ladebefehle ist gewährleistet, daß die entsprechenden "vorbestellten" Daten bereits im jeweils nachzuladenden Cache-Speicher vorliegen, wenn der Prozessor auf sie zugreifen möchte. Dabei kann das Nachladen eines der Cache-Speicher beispielsweise gleichzeitig zu einer Datenübertragung zwischen dem anderen Cache-Speicher und dem Prozessor über den zweiten Datenbus erfolgen.

Wenn ein Cache-Miss auftritt, muß der Prozessor grundsätzlich angehalten werden (wobei die Verarbeitung des jeweils im Befehlsregister geladenen Programmbefehls unterbrochen wird), bis der Cache-Speicher wieder nachgeladen worden ist. Bei der Erfindung werden aufgrund der Cache-Ladebefehle Cache-Misses vermieden oder zumindest in ihrer Anzahl reduziert, so daß ein Anhalten des Prozessors entsprechend vermieden wird.

Der Cache-Ladebefehl ist innerhalb der Programmdatei zwischen die einzelnen durch den Prozessor zu verarbeitenden Programmbefehle eingefügt. Dabei erfolgt eine Verarbeitung der Programmbefehle nach einem Laden in ein Programmbefehlsregister des Prozessors durch den Prozessor, während die Cache-Ladebefehle an die Cache-Steuerung übermittelbar sind, die wiederum entsprechend dem Cache-Ladebefehl einen der Cache-Speicher mit den zukünftig benötigten Daten nachlädt. Es ist also erfindungsgemäß vorgesehen, bei der Programmierung des Prozessors außer den üblichen Programmbefehlen die erfindungsgemäßen Cache-Ladebefehle mit zu programmieren und diese in einer Weise innerhalb des abzuarbeitenden Programmes anzuordnen, daß zukünftig benötigte Daten bereits in einen der Cache-Speicher nachladbar sind, während der Prozessor noch aktuelle Daten aus dem anderen Cache-Speicher ausliest oder Daten in diesen hineinschreibt.

Eine Weiterbildung der Erfindung sieht vor, daß die Programmdatei wenigstens einen Cache-Schreibbefehl enthalten, der ebenfalls an die Cache-Steuerung übermittelt wird und in entsprechender Weise die Übertragung von Daten, die bereits vom Prozessor in einen der Cache-Speicher geschrieben wurden, von diesem über den ersten Datenbus in den Datenspeicher steuert. Somit ist es möglich, den Zeitpunkt des Einschreibens in den Datenspeicher zu bestimmen, so daß z. B. zuvor der erste Datenbus für eine Datenübertragung zwischen dem Datenspeicher und dem jeweils anderen Cache-Speicher zur Verfügung steht.

Eine weitere Weiterbildung der Erfindung sieht vor, daß neben dem Cache-Ladebefehl und/oder dem Cache-Schreibbefehl auch wenigstens ein Cache-Wechselbefehl in entsprechender Weise zwischen den Programmbefehlen des abzuarbeitenden Programmes angeordnet ist. Der Cache-Wechselbefehl ist ebenfalls nach seinem Einlesen in den Prozessor an die Cache-Steuerung übermittelbar und teilt dieser mit, daß ein Wechsel der Cache-Speicher bei der Datenübertragung vom ersten Datenbus zum zweiten Datenbus bzw. umgekehrt erfolgen soll.

Besonders günstig ist es, wenn der Cache-Ladebefehl und/oder der Cache-Schreibbefehl und/oder der Cache-Wechselbefehl gleichzeitig mit einem der Programmbefehle vom Prozessor eingelesen wird. In dem Fall muß der Prozessor die Abarbeitung der Programmbefehle nicht unterbrechen, woraus ein Geschwindigkeitsvorteil resultiert.

Eine Ausführungsform der Erfindung sieht vor, daß im

Datenspeicher Operanden gespeichert sind, während die Programmdatei dem Prozessor über einen Programmdateieingang zuführbar sind.

Eine andere Ausführungsform der Erfindung sieht vor, daß die Programmdatei selbst im Datenspeicher gespeichert sind (der dann der Programmspeicher ist), so daß über die Cache-Ladebefehle bzw. die Cache-Schreibbefehle die Cache-Steuerung für den Programmspeicher des Prozessors gesteuert wird.

Die Erfindung ermöglicht eine echtzeitfähige Datenübertragung zwischen Datenspeicher und Prozessor und einen unterbrechungsfreien kontinuierlichen Datenfluß, so daß die Arbeit des Prozessors nicht unterbrochen werden muß. Daher eignet sich die Erfindung besonders zum Einsatz in Schaltungsanordnungen mit digitalen Signalprozessoren.

Nachfolgend wird die Erfindung anhand der in den Figuren der Zeichnung dargestellten Ausführungsbeispielen näher erläutert.

Fig. 1 zeigt ein Ausführungsbeispiel der Erfindung:

Fig. 2A bis 2D zeigen vier verschiedene Betriebszustände für das Ausführungsbeispiel nach Fig. 1.

Fig. 1 zeigt eine Schaltungsanordnung mit einem Prozessor DSP, einem Datenspeicher MEM und zwei Cache-Speichern C1, C2. Dargestellt sind nur die für die Erfindung wesentlichen Komponenten. Es können aber auch andere Komponenten in beliebiger Weise hinzugefügt werden. Der Prozessor DSP weist Anschlüsse P/O auf, denen im Datenspeicher MEM gespeicherte Daten zuzuführen sind oder über die vom Prozessor DSP im Datenspeicher MEM zu speichernde Daten ausgebar sind, um anschließend einer Verarbeitung durch den Prozessor unterzogen zu werden. Um eine Anpassung der durch den Datenspeicher MEM erreichbaren Datenrate an die Verarbeitungsgeschwindigkeit des Prozessors zu erreichen, sind die Cache-Speicher C1, C2 vorgesehen. Die im Datenspeicher MEM gespeicherten Daten sind über Anschlüsse MEM/O desselben aus ihm auslesbar. Im Datenspeicher MEM vom Prozessor DSP zu speichernde Daten sind ebenfalls über die Anschlüsse MEM/O einlesbar. Die Anschlüsse MEM/O des Datenspeichers MEM sind über einen ersten Datenbus D1 sowohl mit entsprechenden ersten Anschlüssen C1/O1 des ersten Cache-Speichers C1, als auch mit entsprechenden ersten Anschlüssen C2/O1 des zweiten Cache-Speichers C2 verbunden. Zweite Anschlüsse C1/O2, C2/O2 der beiden Cache-Speicher C1, C2 sind auf entsprechende Weise über einen zweiten Datenbus D2 mit den Anschlüssen P/O des Prozessors DSP verbunden. Somit sind Daten, die im Datenspeicher MEM gespeichert sind, wahlweise über den ersten Cache-Speicher C1 oder den zweiten Cache-Speicher C2 zu den Anschlüssen P/O des Prozessors DSP übertragbar und umgekehrt. Beim geschilderten Ausführungsbeispiel sind die Anschlüsse des Datenspeichers MEM, der Cache-Speicher C1, C2 und des Prozessors DSP jeweils Ein- und Ausgänge und die Datenbusse D1 und D2 bidirektional.

Die Erfindung weist also zwei parallel geschaltete Cache-Speicher C1, C2 auf.

Die Schaltungsanordnung in Fig. 1 weist außerdem eine Cache-Steuerung CTR auf, die über entsprechende Adressbusse C1ADR, C2ADR, MEMADR sowohl jeden einzelnen Cache-Speicher C1, C2, als auch den Datenspeicher MEM adressieren kann.

Vom Prozessor DSP werden über entsprechende Leitungen die Adressen ADR von aktuell durch den Prozessor DSP benötigten Daten und die Adressen ADR, an denen von ihm erzeugte Daten im Datenspeicher MEM gespeichert werden sollen, an die Cache-Steuerung CTR übertragen. Ob Daten durch den Prozessor gelesen oder geschrieben werden sollen, wird der Cache-Steuerung CTR über ein durch den

Prozessor DSP erzeugt Schreibsignal W und ein Lesesignal L, mitgeteilt, das beispielsweise durch ein Bit realisiert sein kann, daß Bestandteil der an die Cache-Steuerung CTR übermittelten Adressen ADR ist.

Weiterhin werden der Cache-Steuerung CTR vom Prozessor DSP die Adressen zukünftig benötigter Daten in Form von Cache-Ladebefehlen CL übermittelt, die weiter unten noch erläutert werden. Zukünftig benötigte Daten sind solche, die nicht als nächstes, sondern erst zu einem späteren Zeitpunkt vom Prozessor DSP aufgrund eines erst dann eingelesenen Programmbefehls PDC verarbeitet werden.

Die aktuelle Datenanforderung ist jeweils die Adresse des als nächstes benötigten Datums innerhalb des Datenspeichers MEM. Die Cache-Steuerung CTR führt eine entsprechende Adressierung desjenigen Cache-Speichers C1, C2 durch, in dem sich das Datum mit dieser Adresse befindet.

Die Cache-Ladebefehle CL sind innerhalb von in einem Programmspeicher PM der Schaltungsanordnung gespeicherten Programmdateien PD, die ein durch den Prozessor DSP abzuarbeitendes Programm bilden, angeordnet. Die Programmdateien PD enthalten außer den Cache-Ladebefehlen CL Programmbefehle PDC, die wie bei herkömmlichen Prozessoren gestaltet sind. Außerdem enthalten die Programmdateien PD Cache-Schreibbefehle CW und Cache-Wechselbefehle CC, auf die weiter unten noch eingegangen wird.

Beim gezeigten Ausführungsbeispiel in Fig. 1 verfügt der Prozessor DSP über Programmdateieneingänge PIN, die zusätzlich zu den Eingängen IN vorhanden sind. Eine solche Architektur mit getrennten Programmdatei- und Operandeneingängen liegt insbesondere bei digitalen Signalprozessoren vor. Den Programmdateieneingängen PIN sind die Programmdateien PD über einen Programmdateibus vom Programmspeicher PM zuführbar. Die Programmdateien PD werden sequentiell gemäß einem Arbeitstakt CLK des Prozessors DSP über die Programmdateieneingänge PIN eingelesen. Dabei werden die Programmbefehle PDC in ein Programmbefehlsregister REG eingelesen und anschließend einer Bearbeitung durch den Prozessor DSP unterzogen. Dagegen werden die Cache-Ladebefehle CL, die Cache-Schreibbefehle CW und die Cache-Wechselbefehle CC vom Prozessor DSP an die Cache-Steuerung CTR weitergeleitet.

Der Programmspeicher PM hat wahlweise einen eigenen Programm-Cache-Speicher, der in Fig. 1 allerdings nicht separat dargestellt ist.

In Fig. 1 ist durch den gestrichelten Kasten angedeutet, daß die Cache-Steuerung CTR selbstverständlich auch Bestandteil des Prozessors DSP sein kann. Auch werden ihr die Cache-Ladebefehle CL, die Cache-Schreibbefehle CW und die Cache-Wechselbefehle CC bei anderen Ausführungsformen der Erfindung direkt vom Programmspeicher PM zugeführt, ohne daß diese zuvor vom Prozessor DSP eingelesen werden.

In Fig. 1 sind innerhalb des Programmspeichers PM beispielhaft einige der Programmdateien PD eingetragen. Dabei sollen die Programmbefehle PDC, die Cache-Wechselbefehle CC, die Cache-Schreibbefehle CW und die Cache-Ladebefehle CL von oben nach unten adressenmäßig aufeinanderfolgend angeordnet sein. Die Adressierung der Programmdateien PD erfolgt durch den Prozessor DSP über einen Programmadreßbus PADR. Bei der Abarbeitung dieses Programms lädt der Prozessor DSP zunächst den ersten Programmbefehl PDC in das Programmbefehlsregister REG und verarbeitet diesen. Anschließend geschieht dasselbe mit dem zweiten Programmbefehl PDC. Im nächsten Zyklus des Arbeitstaktes CLK liest der Prozessor DSP über seine Programmdateieneingänge PIN den Cache-Ladebefehl CL ein und leitet diesen unbearbeitet an die Cache-Steuerung CTR

weiter. Die Cache-Steuerung CTR führt ein dem Cache-Ladebefehl CL entsprechendes Nachladen eines der beiden Cache-Speicher C1, C2 durch, wie anhand der Fig. 2A und 2D weiter unten noch erläutert wird.

Es ist vorteilhaft, aber nicht zwingend, wenn bei der Anordnung nach Fig. 1 der Prozessor DSP zeitgleich (d. h. innerhalb desselben Zyklus seines Arbeitstaktes CLK) mit dem Cache-Ladebefehl CL auch den adressenmäßig nachfolgenden oder den vorhergehenden Programmbefehl PDC in sein Programmbefehlsregister REG lädt. Er kann daher nämlich unabhängig von der Übermittlung der Cache-Ladebefehle CL an die Cache-Steuerung CTR mit der Abarbeitung der Programmbefehle PDC fortfahren. Dasselbe gilt für den Cache-Schreibbefehl CW und den Cache-Wechselbefehl CC. Für diese gleichzeitige Laden ist es erforderlich, daß die entsprechenden Befehle CL, CW, CC, PDC durch eine gemeinsame Adresse über den Programmadreßbus PADR adressierbar sind.

Die Cache-Ladebefehle CL enthalten günstigerweise eine Startadresse, ab der ein Datenblock definierter Größe aus dem Datenspeicher MEM durch die Cache-Steuerung CTR in den jeweils gerade nachzuladenden der Cache-Speicher C1, C2 zu übertragen ist. Es ist auch möglich, daß der Cache-Ladebefehl CL darüber hinausgehende Informationen aufweist, wie beispielsweise über die Anzahl von aufeinanderfolgenden Datenblöcken, die aufgrund eines gerade durch die Cache-Steuerung CTR auszuführenden Cache-Ladebefehls CL nachgeladen werden sollen. Außerdem kann der Cache-Ladebefehl auch eine Modulo-Information enthalten, bei der innerhalb von aufeinanderfolgenden zu ladenden Datenblöcken aufgrund der Startadresse das Nachladen mit einem beliebigen dieser Datenblöcke begonnen werden kann. Die erwähnten Datenblöcke weisen mehrere Daten mit aufeinanderfolgenden Adressen auf. Somit kann mit einem sehr kurzen Cache-Ladebefehl CL, der nur wenige Bit aufweist, ein Nachladen einer großen Menge von Daten ausgelöst werden.

Da aufgrund des Cache-Ladebefehls CL immer Datenblöcke definierter Größe zwischen Datenspeicher MEM und Cache-Speicher C1, C2 übertragen werden, ist die Zeit zur Ausführung dieser Datenübertragung bekannt. Da außerdem bekannt ist, wieviele Zyklen des Arbeitstaktes CLK der Prozessor DSP benötigt, um jeweils einen der Programmbefehle PDC abzuarbeiten (im allgemeinen geschieht dies innerhalb eines Zyklus), ist der Cache-Ladebefehl CL innerhalb der Programmdateien PD günstigerweise so angeordnet, daß die durch ihn für die zukünftige Abarbeitung angeforderten Daten rechtzeitig im jeweiligen Cache-Speicher C1, C2 bereitgestellt werden. Rechtzeitig bedeutet dabei, bevor einer der Programmbefehle PDC einen Zugriff auf diese angeforderten Daten erforderlich macht.

Die im Programmspeicher PM in Fig. 1 gespeicherten Programmdateien PD weisen auch wenigstens einen der Cache-Wechselbefehle CC auf. Dieser wird ebenfalls nicht ins Programmbefehlsregister REG des Prozessors eingelesen, sondern von diesem an die Cache-Steuerung CTR weitergeleitet. Der Cache-Wechselbefehl CC teilt der Cache-Steuerung CTR mit, daß bei beiden Cache-Speichern C1, C2 ein Wechsel bezüglich der Nutzung des ersten D1 und zweiten D2 Datenbusses erfolgen soll. Aufgrund des Cache-Wechselbefehls CC bewirkt die Cache-Steuerung CTR einen solchen Wechsel. Innerhalb der Programmdateien PD ist der Cache-Wechselbefehl CC adressenmäßig so eingefügt, daß ein Nachladen des entsprechenden Cache-Speichers C1, C2 aufgrund eines vorhergehenden Cache-Ladebefehls CL bereits abgeschlossen ist und das nächste durch den Prozessor DSP anzufordernde Datum sich unter diesen nachgeladenen Daten befindet, bevor der Cache-Wechselbefehl CC von der

Cache-Steuerung CTR ausgeführt wird. Die Cache-Wechselbefehl CC kommen immer dann zum Einsatz, wenn eine Datenübertragung zwischen Prozessor DSP und Datenspeicher MEM zu erfolgen hat und die entsprechenden Daten bereits über einen der Datenbusse D1, D2 in einen der Cache-Speicher C1, C2 übertragen wurden, so daß anschließend über den jeweils anderen der Datenbusse die weitere Datenübertragung zum Zielort (Prozessor DSP oder Datenspeicher MEM) erfolgen kann.

In entsprechender Weise wie der Cache-Ladebefehl CL und der Cache-Wechselbefehl CC ist der Cache-Schreibbefehl CW aus dem Programmspeicher PM über die programmanschlüsse PIN des Prozessors DSP an die Cache-Steuerung CTR übermittelbar. Beim gezeigten Ausführungsbeispiel erfolgt sowohl ein Lese-Caching (zu dessen Optimierung der Cache-Lesebefehl CL in der bereits beschriebenen Weise dient) als auch ein Schreib-Caching. Der Prozessor DSP speichert also Daten im Datenspeicher MEM, die zuvor durch die Cache-Speicher C1, C2 zwischengespeichert werden. Dabei erfolgt die Datenübertragung zwischen Prozessor DSP zu einem der Cache-Speicher C1, C2 in üblicher Weise durch Übermittlung der gewünschten Speicheradresse ADR und eines Schreibsignals W an die Cache-Steuerung CTR. Diese nimmt jedoch die weitere Übertragung dieser Daten vom Cache-Speicher C1, C2 zum Datenspeicher MEM nicht automatisch vor, sondern erst, wenn sie einen entsprechenden Cache-Schreibbefehl CW erhält. Erst der Cache-Schreibbefehl CW bewirkt also eine Übertragung der vom Prozessor DSP bereits in einen der Cache-Speicher C1, C2 geschriebenen Daten von dort in den Datenspeicher MEM. Somit ist der Zeitpunkt dieser Datenübertragung mittels des Cache-Schreibbefehls CW festlegbar. Der Zugriff der beiden Cache-Speicher C1, C2 auf die beiden Datenbusse D1, D2 kann somit optimiert werden.

Ebenso wie beim Cache-Ladebefehl CL, ist es günstig, daß der Cache-Wechselbefehl CC und der Cache-Schreibbefehl CW gleichzeitig mit einem vorhergehenden oder nachfolgenden Programmbefehl PDC vom Prozessor DSP einlesbar ist, so daß dieser den Programmbefehl PDC gleichzeitig mit der Übermittlung des Cache-Wechselbefehls CC bzw. des Cache-Schreibbefehls CW an die Cache-Steuerung CTR in sein Programmbefehlsregister REG aufnimmt.

In einer weiter vereinfachenden Darstellung in den Fig. 2A bis 2D sind verschiedene, aufeinander folgende Betriebszustände der Schaltungsanordnung aus Fig. 1 dargestellt.

Dabei enthält der Datenspeicher MEM in Fig. 2A an aufeinanderfolgenden Adressen die Daten D1 bis D10. In den ersten Cache-Speicher C1 sind (ausgelöst durch einen vorhergehenden Cache-Ladebefehl CL) bereits die Daten D1 bis D3 geladen, während im zweiten Cache-Speicher C2 beliebige Daten D0 enthalten sind, die z. B. bereits vom Prozessor DSP gelesen wurden. Im aktuell gezeigten Betriebszustand sind durch die Cache-Steuerung CTR aufgrund der Adressen der aktuellen Datenanforderungen des Prozessors DSP und aufgrund des Lesesignals L die Daten D1 bis D3 über den zweiten Datenbus D2 aus dem ersten Cache-Speicher C1 in den Prozessor DSP einlesbar. Gleichzeitig sind durch die Cache-Steuerung CTR aufgrund eines der Cache-Ladebefehle CL über den ersten Datenbus D1 Daten aus dem Datenspeicher MEM in den zweiten Cache-Speicher C2 einlesbar. Angedeutet ist diese zeitweilige Zuordnung der Datenbusse D1, D2 zu den Cache-Speichern C1, C2 in den Fig. 2A bis 2D einerseits durch die durchgezogenen, und andererseits durch die gestrichelten Linien, die die Datenbusse D1, D2 darstellen.

Fig. 2B zeigt den Zustand, nachdem aufgrund eines der

Cache-Ladebefehle CL die Daten D4 bis D6 über den ersten Datenbus D1 in den zweiten Cache-Speicher C2 übertragen wurden. Außerdem sind Daten D11 bis D13 vom Prozessor DSP über den zweiten Datenbus D2 in den ersten Cache-Speicher C1 übertragen worden, wozu der Prozessor DSP das Schreibsignal W und die entsprechenden Adressen ADR für die zu speichernden Daten an die Cache-Steuerung CTR übertragen hat.

Durch einen der Cache-Wechselbefehle CC erfolgt nun gemäß Fig. 2C ein Wechsel der Cache-Ansteuerung in der Weise, daß nun durch den Prozessor DSP über den zweiten Datenbus D2 auf die Daten D4 bis D6 im zweiten Cache-Speicher C2 aufgrund der an die Cache-Steuerung CTR übermittelten Adressen ADR der aktuell benötigten Daten und des Lesesignals L zugreifbar ist. Gleichzeitig werden die im ersten Cache-Speicher C1 vorhandenen Daten D11 bis D13 aufgrund eines der Cache-Schreibbefehle CW über den ersten Datenbus D1 im Datenspeicher MEM abgelegt.

Fig. 2D zeigt nun den Zustand, nachdem gleichzeitig über einen der Cache-Ladebefehle CL angeforderte, zu einem späteren Zeitpunkt benötigte Daten D8 bis D10 über den ersten Datenbus D1 durch die Cache-Steuerung CTR in den ersten Cache-Speicher C1 nachgeladen wurden und weitere Daten D14 bis D16 vom Prozessor DSP aufgrund der Adressen ADR und des Schreibsignals W über den zweiten Datenbus D2 zum zweiten Cache-Speicher C2 übertragen wurden. Durch einen weiteren Cache-Wechselbefehl CC kann nun erneut die Zuordnung der Datenbusse D1, D2 zu den Cache-Speichern C1, C2 gewechselt werden.

Die anhand der Fig. 2A bis 2D gemachten Erläuterungen machen deutlich, daß aufgrund des erfindungsgemäßen Cache-Ladebefehls CL ein unterbrechungsfreier Datenfluß zwischen dem Datenspeicher MEM und dem Prozessor DSP realisierbar ist. Cache-Misses können vollständig vermieden werden. Außerdem ermöglicht der erfindungsgemäße Cache-Schreibbefehl CW vorteilhaft ein von Zugriffen des Prozessors DSP auf einen der Cache-Speicher C1, C2 unabhängiges Speichern von Daten aus dem jeweils anderen Cache-Speicher in den Datenspeicher MEM. Durch beide Maßnahmen wird die Datenübertragung zwischen Datenspeicher MEM und Prozessor DSP beschleunigt und verstetigt.

Die Cache-Steuerung CTR wird bevorzugt als verdrahtete Logik realisiert. Sie kann jedoch auch durch einen zweiten Prozessor realisiert sein, der zusätzlich zum Prozessor DSP vorhanden ist.

Anders als beim geschilderten Ausführungsbeispiel sind andere Ausgestaltungen der Erfindung denkbar, bei denen wahlweise nur ein Schreib- oder nur ein Lese-Caching (beispielsweise, wenn der Datenspeicher identisch mit dem Programmspeicher ist) durchgeführt wird. Entsprechend sind dann nur die Cache-Schreibbefehle CW oder die Cache-Lesebefehle CL für ihre Steuerung notwendig.

Anders als beim Ausführungsbeispiel in Fig. 1 gezeigt, ist es auch möglich, daß die Anschlüsse P/O des Prozessors DSP die Programmdateneingänge PIN sind. Dann ist der Datenspeicher MEM der Programmspeicher PM und die beiden Cache-Speicher C1, C2 dienen der Zwischenspeicherung der Programmdaten PD. Damit würde über die Cache-Ladebefehle CL und die Cache-Wechselbefehle CC die Steuerung des Zwischenspeicherns von Programmdaten PD in den Cache-Speichern C1, C2 erfolgen. Dagegen zeigt die Fig. 1 ein Ausführungsbeispiel der Erfindung, bei dem im Datenspeicher MEM Operanden speicherbar sind, die aufgrund der Programmbefehle PDC für eine Verarbeitung durch den Prozessor DSP erforderlich sind.

In Abwandlung des geschilderten Ausführungsbeispiels, bei dem die ersten C1/O1, C2/O1 und zweiten C1/O2,

C2I/O2 Anschlüsse jedes Cache-Speichers C1, C2 getrennt voneinander sind (es handelt sich um sogenannte Dual-Port-Speicher), ist es möglich, daß sie für jeden Cache-Speicher identisch miteinander sind (sogenannte Single-Port-Speicher). Dann hat jeder Cache-Speicher C1, C2 nur eine Art von Anschlüssen, die jeweils sowohl mit dem ersten Datenbus D1 als auch mit dem zweiten Datenbus D2 verbunden sind. Haben die Datenbusse D1, D2 eine unterschiedliche Breite, können die ersten und zweiten Anschlüsse nur teilweise identisch sein.

Patentansprüche

1. Schaltungsanordnung mit
 - einem Prozessor (DSP) und einem Datenspeicher (MEM) mit jeweils Anschlüssen (PI/O, MEMI/O),
 - mit zwei Cache-Speichern (C1, C2) mit ersten Anschlüssen (C1I/O1, C2I/O1) und zweiten Anschlüssen (C1I/O2, C2I/O2)
 - einem ersten Datenbus (D1), über den jeder der Anschlüsse (MEMI/O) des Datenspeichers (MEM) mit je einem der ersten Anschlüsse (C1I/O1, C2I/O2) jedes Cache-Speichers (C1, C2) verbunden ist und
 - einem zweiten Datenbus (D2), über den jeder der Anschlüsse (PI/O) des Prozessors (DSP) mit je einem der zweiten Anschlüsse (C1I/O2, C2I/O2) jedes Cache-Speichers (C1, C2) verbunden ist.
2. Schaltungsanordnung nach Anspruch 1, bei der eine Datenübertragung gleichzeitig über den ersten Datenbus (D1) zwischen einem der Cache-Speicher (C1, C2) und dem Datenspeicher (MEM) und über den zweiten Datenbus (D2) zwischen dem anderen Cache-Speicher (C1, C2) und dem Prozessor (DSP) erfolgt.
3. Schaltungsanordnung nach einem der vorstehenden Ansprüche mit zusätzlich
 - einer Cache-Steuerung (CTR), die durch entsprechende Adressierung (C1ADR, C2ADR, MEMADR) der Cache-Speicher (C1, C2) und des Datenspeichers (MEM) die Datenübertragung auf dem ersten (D1) und dem zweiten (D2) Datenbus steuert,
 - einem Programmspeicher (PM) zum Speichern von Programmdaten (PD), die Programmbefehle (PDC) enthalten, die zur Verarbeitung durch den Prozessor (DSP) über Programmdateneingänge (PIN) in ein Befehlsregister (REG) geladen werden,
 - wobei die Programmdaten (PD) außerdem wenigstens einen Cache-Ladebefehl (CL) zur Übermittlung an die Cache-Steuerung (CTR) enthalten, die nach dessen Erhalt eine entsprechende Datenübertragung vom Datenspeicher (MEM) über den ersten Datenbus (D1) zu einem der Cache-Speicher (C1, C2) bewirkt, und
 - wobei die Programmbefehle (PDC) und der Cache-Ladebefehl (CL) so innerhalb der Programmdaten (PD) adressenmäßig aufeinanderfolgend angeordnet sind, daß durch den Cache-Ladebefehl (CL) bereits Daten angefordert werden, bevor sie für eine durch einen der Programmbefehle (PDC) ausgelöste Verarbeitung vom Prozessor (DSP) benötigt werden.
4. Schaltungsanordnung nach Anspruch 3, bei der
 - über den zweiten Datenbus (D2) Daten vom Prozessor (DSP) in die Cache-Speicher (C1, C2)

übertragen werden und

- die Programmdaten (PD) zusätzlich oder alternativ zum Cache-Ladebefehl (CL) wenigstens einen Cache-Schreibbefehl (CW) zur Übermittlung an die Cache-Steuerung (CTR) enthalten, die nach dessen Erhalt eine entsprechende Übertragung der vom Prozessor (DSP) im jeweiligen Cache-Speicher (C1, C2) gespeicherten Daten über den ersten Datenbus (D1) zum Datenspeicher (MEM) bewirkt.

5. Schaltungsanordnung nach einem der Ansprüche 3 oder 4, bei der

- die Programmdaten (PD) außerdem wenigstens einen Cache-Wechselbefehl (CC) zur Übermittlung an die Cache-Steuerung (CTR) enthalten und
- aufgrund des Cache-Wechselbefehls (CC) die Cache-Steuerung (CTR) bei beiden Cache-Speichern (C1, C2) einen Wechsel der Datenübertragung vom ersten Datenbus (D1) zum zweiten Datenbus (D2) bzw. umgekehrt bewirkt.

6. Schaltungsanordnung nach einem der Ansprüche 3 bis 5, bei der

- die Verarbeitung der Programmbefehle (PDC) durch den Prozessor (DSP) gemäß einem Takt (CLK) des Prozessors erfolgt und
- innerhalb desselben Taktzyklus eine Übermittlung des Cache-Ladebefehls (CL) und/oder des Cache-Schreibbefehls (CW) und/oder des Cache-Wechselbefehls (CC) vom Programmspeicher (PM) an die Cache-Steuerung (CTR) und ein Laden eines der Programmbefehle (PDC) in das Befehlsregister (REG) erfolgt.

7. Schaltungsanordnung nach einem der Ansprüche 3 bis 6, bei der

- der Programmspeicher (PM) mit dem Datenspeicher (MEM) identisch ist und
- die Anschlüsse (PI/O) des Prozessors (DSP) die Programmdateneingänge sind.

8. Schaltungsanordnung nach einem der Ansprüche 3 bis 6, bei der

- im Datenspeicher (MEM) Operanden gespeichert sind,
- die Anschlüsse (PI/O) des Prozessors (DSP) Operandeneingänge sind und
- der Prozessor (DSP) Programmdateneingänge (PIN) hat, denen die Programmdaten (PD) zuführbar sind.

9. Schaltungsanordnung nach einem der vorstehenden Ansprüche, bei der der Prozessor (DSP) ein digitaler Signalprozessor ist.

10. Schaltungsanordnung nach einem der vorstehenden Ansprüche, bei der wenigstens ein Teil der ersten Anschlüsse (C1I/O1, C2I/O1) wenigstens eines der Cache-Speicher (C1, C2) mit wenigstens einem Teil seiner zweiten Anschlüsse (C1I/O2, C2I/O2) identisch ist.

Hierzu 5 Seite(n) Zeichnungen

- Leerseite -

THIS PAGE BLANK (USPTO)

FIG 1

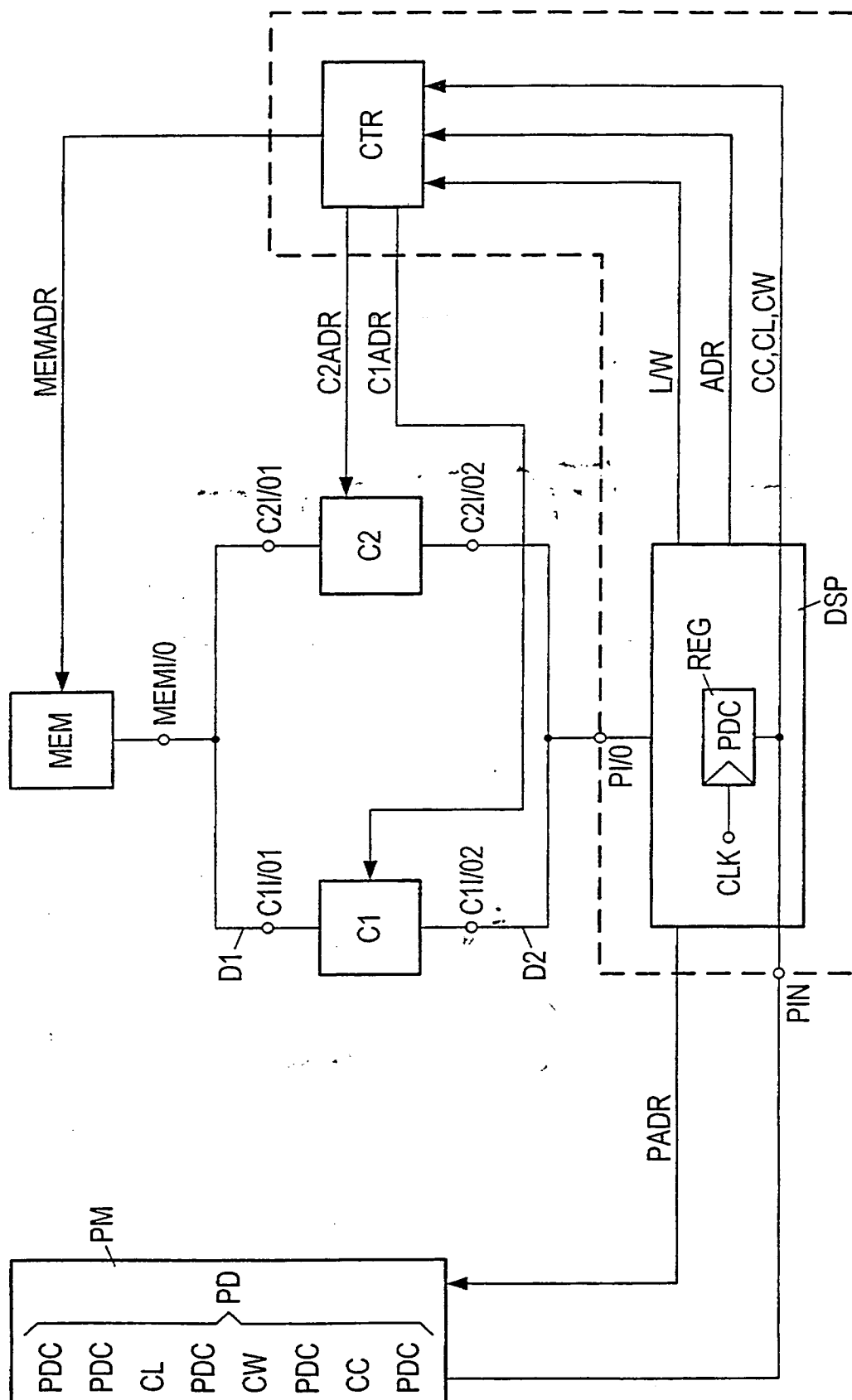


FIG 2A

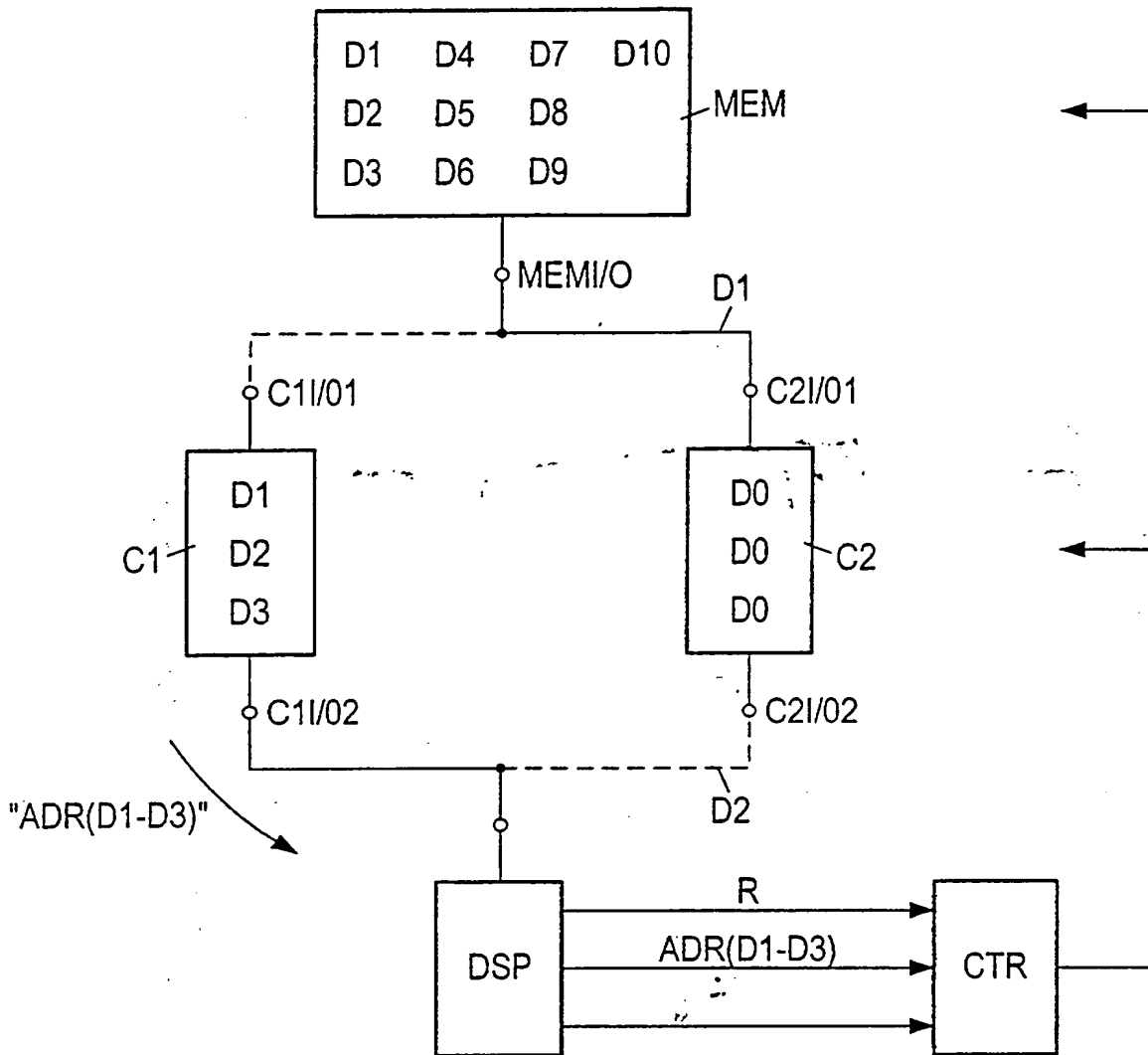


FIG 2B

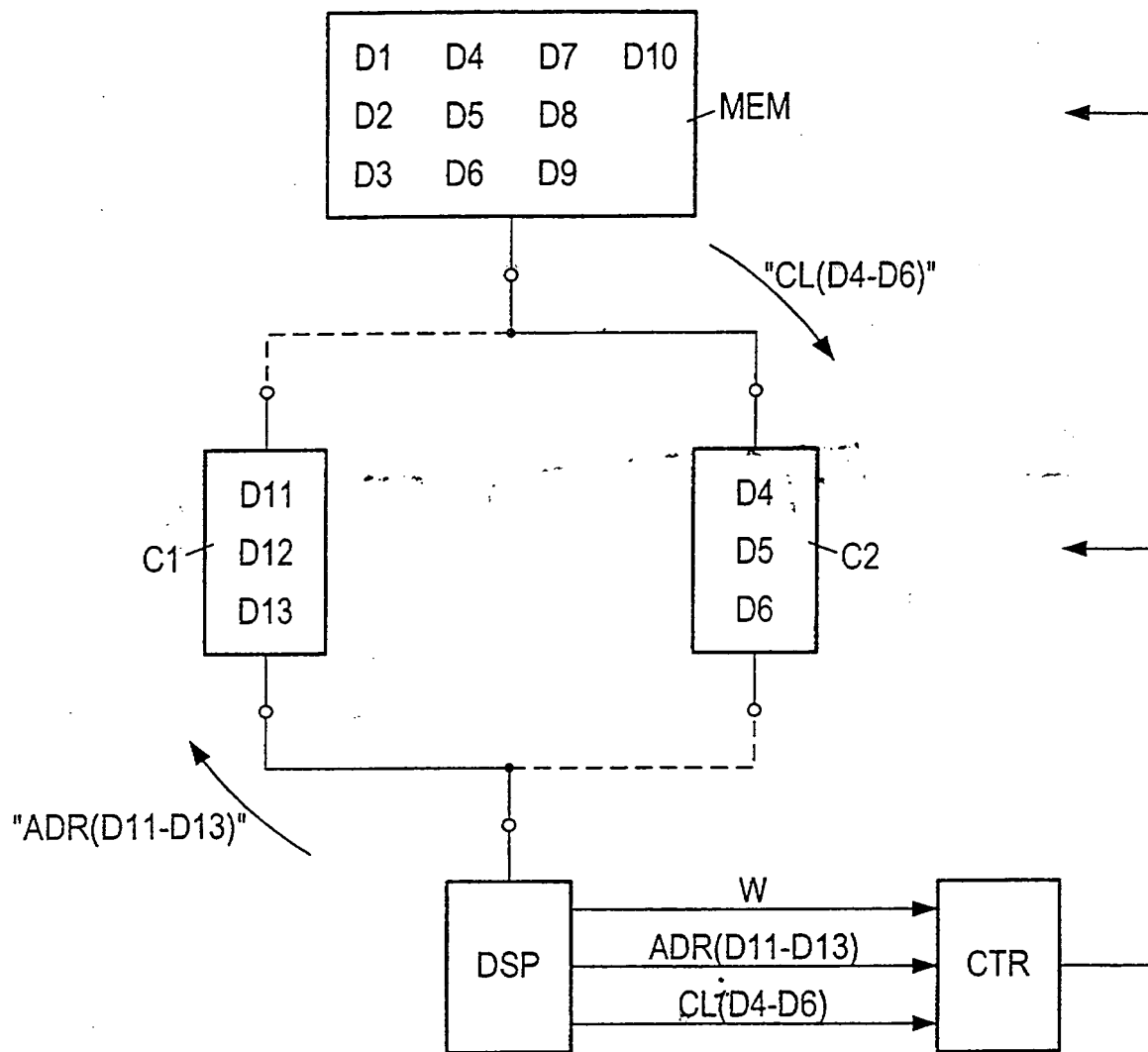


FIG 2C

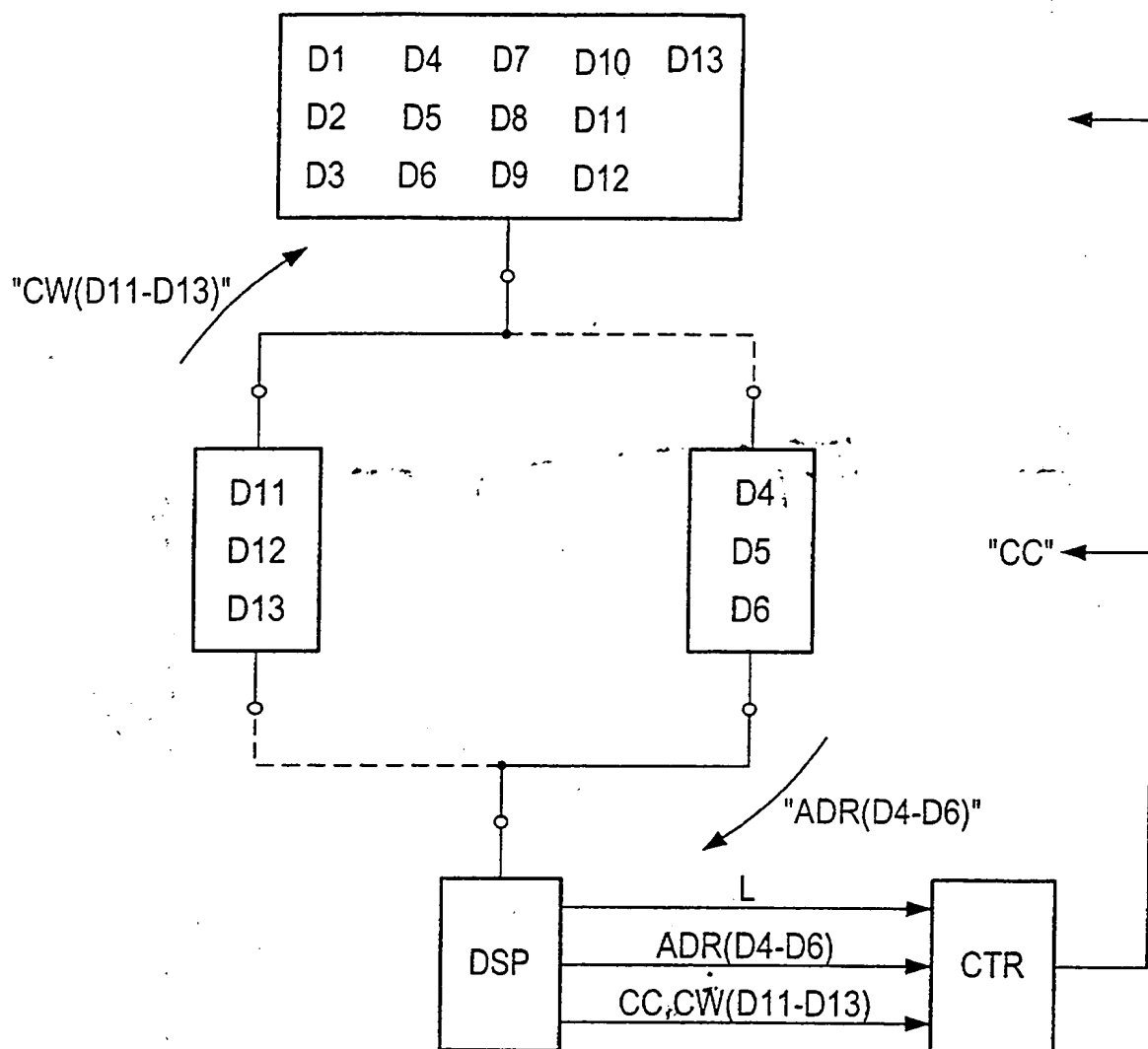


FIG 2D

